MULTIDISCIPLINARY DESIGN PROJECT
PROJECT MANAGEMENT PLAN

Version 1.5

1st February 2021

GROUP 9

BHATIA RITIK
DATTA ANUSHA
HOY DI SHENG MAX
IRVIN TING SIE ZE
JEREMY LEE KIAN KIAT
KANG WEI BIN
NUR INSYIRAH LUKEMAN
QUAH DIAN WEI

SUPERVISORS

Lam Siew Kei
Owen Newton
Bector Ridhima
Chen Change Loy

TABLE OF CONTENTS

Content	Page
Version History	3
Introduction ● Project Objectives	4
Scope	4
Assumptions and Constraints	5
Work Breakdown Structure	6
Project Organisation	10
Approach	12
Risk Management	13
Schedule/Time Management	16
Project Schedule - Gantt Chart	17
Appendix: Key Terms	18

VERSION HISTORY

Version	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	Datta Anusha & Bhatia Ritik	24/01/2021	Bhatia Ritik	24/01/2020	Added Project Objectives, Project Scope and Work Breakdown Structure
1.1	Irvin Ting & Dian Wei	28/01/2021	Bhatia Ritik	29/01/2020	Added risk management, updated RPi WBS
1.2	Jeremy Lee & Wei Bin	28/01/2021	Bhatia Ritik	29/01/2020	Added WBS Diagram, updated Android WBS, editing of resources constraint
1.3	Bhatia Ritik & Datta Anusha	29/01/2021	Bhatia Ritik	29/01/2020	Update WBS, schedule of project
1.4	Wei Bin	29/01/2021	Bhatia Ritik	30/01/2020	Add Gantt Chart for WBS
1.5	Datta Anusha	01/02/2021	Bhatia Ritik	01/02/2021	Final edits & formatting of proposal

INTRODUCTION

PROJECT OBJECTIVES

MDP stands for **Multidisciplinary Design Project** and it is a collaborative course consisting of randomly allocated teams of 7-8 people, with students from Computer Science, Computer Engineering, BCG and BCE. This project aims to provide students with the holistic skill of collaborating with people of diverse backgrounds, working together and combining skills from different streams to create a functional, autonomous robot, capable of performing many different tasks. The main functionalities that the resulting robot should cover are as follows:

- The robot must be able to **autonomously navigate** an unexplored, unknown area as and when indicated. It should be capable of **avoiding the obstacles** in the path during this phase and identify the start and the end points for the exploration task.
- The robot should make use of Raspberry Pi and the sensors to detect images from a set of known images, while exploring the unknown area. It should be able to **correctly classify** the image and display its coordinates.
- The robot should be able to utilize appropriate algorithms to reach from a start point to an end point by traversing the **fastest path** possible.

The students are also expected to establish **Bluetooth communication** between the Nexus 7 Android tablet and the Raspberry Pi. The Arduino has all the motors, sensors and the Raspberry Pi to communicate with. Hence, the Raspberry Pi stands as the connection point for all the interfaces and ensures proper communication and coordination between each.

Apart from the above technical implementations, MDP aims to equip students with the necessary **soft skills** like teamwork, communication, healthy debates and leadership, which are required to thrive in the corporate world.

SCOPE

The scope of this multidisciplinary project is to develop a robotic system with diverse functionalities like maze exploration, discovering the shortest path traversal, navigating obstacles and correctly identifying images. The integration of four knowledge spheres– Android, Algorithms, Raspberry Pi and Arduino - is required to implement the Android Remote Controller module, Maze Exploration and Traversal module, and Mobile Remote module of the robotic system.

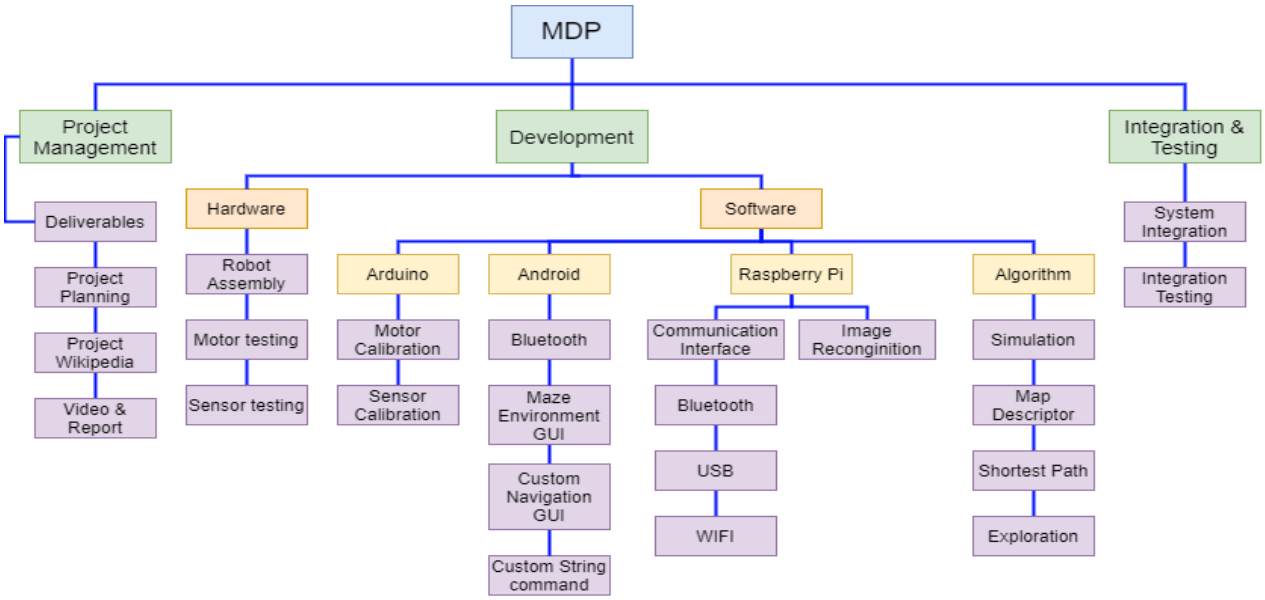
The successful completion of the project shall be measured by the achievement of all the above mentioned functionalities, along with a video report and Media Wiki documentation. The video report shall consist of a brief overview of the contributions of the team towards the project, as well as our journey from the initial design stage to the final execution stage. The Media Wiki documentation shall contain a detailed account of the various components of the robotic system, which will serve to explain our approach and implementation of each of these subsystems.

ASSUMPTIONS & CONSTRAINTS

Assumptions
It is assumed that all group members will carry out their assigned responsibilities and tasks dutifully and sincerely of acceptable standard.
The deliverables and requirements of the project are fixed throughout the duration of MDP.
The system components are in working condition and do not malfunction at any stage of the project. If they do, then replacements are quickly available to minimise downtime.
All the team members do proper research before proceeding with the task and ensure that the quality of all work is up to standard, to avoid unforeseen circumstances in the future.

Constraints	
Time	Collaborating as a team to work on the project as every member has their respective commitment like a part time job.
	Limited hours of free access due to the Covid-19 pandemic which limits the time for calibration and integration.
Resources	The assumption that all components work fine and are intercompatible.
	Limited experience and knowledge as this is the first time for most team members working on a project of such a big scale.
Manpower	The team is limited to materials provided for the project, and prohibited from purchasing any additional materials to improve performance of the robot.

Work Breakdown Structure



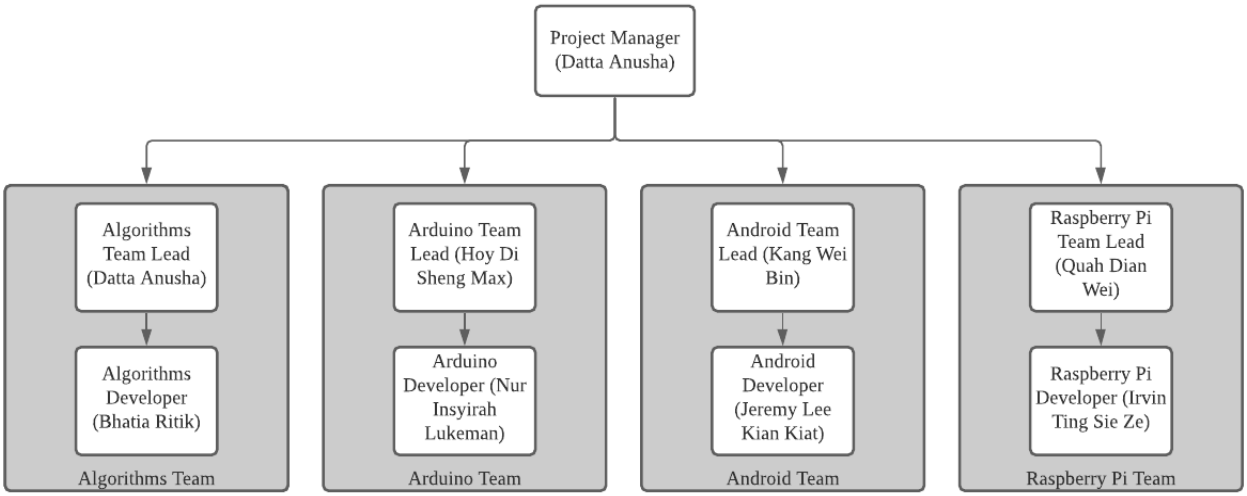
S/N	Work (Activity/Task)	Description	Efforts Estimate	Dependencies
Project Planning/Management (PM1)				
PM1.1	Project Plan Discussion	Discuss with the team to set project goals and objectives, finalise project plan and project schedule.	2 days	-
PM1.2	Delegation of Roles & Responsibilities	All team members are assigned their primary roles and delegated work accordingly.	1 day	PM1.1
PM1.3	Environment Setup	Setup development environment and communication channel for project (GitHub & MS Teams)	1 day	PM1.1
PM1.4	Analyse Project Deliverables	Create a common understanding of assessment criteria within the team, to excel in project deliverables provided on NTUlearn.	1 days	PM1.1

PM1.5	Learning/ Knowledge Acquisition	Conduct extensive research on the relevant development tools and various components of the robotic system to aid project development.	5 days	PM1.4
Robotic System Assembly (RSA2)				
RSA2.1	Hardware Components Collection and Testing	Cross check against hardware checklist to ensure all components have been received, and test to ensure none of them are in faulty condition.	7 days	PM1.4
RSA2.2	Assemble Robot	Build a functional robot by successfully assembling all the required components.	1 day	RSA2.1
Arduino Development (AD3)				
AD3.1	Set up Arduino Environment	Download and set up the Arduino integrated development environment, by importing necessary libraries for the motor and sensors.	1 day	PM1.4
AD3.2	Motor Calibration	Establish correlation of the Input speed to RPM for both robotic motors - without load and with load.	2 days	RSA2
AD3.3	PID of motor	Reconfigure robot to move straight with the help of PID.	3 days	AD3.2
AD3.4	IR Sensor Calibration	Calibrate each sensor to produce readings that are accurate.	5 days	RSA2
AD3.5	Main code for Arduino	Invoke the motor and sensor Arduino codes to allow the robot to accurately move a certain distance or turn at a certain angle.	15 days	PM1.4
Raspberry Pi Development (RP4)				
RP4.1	Raspberry Pi Setup	Setup the RPi system for development.	3 days	PM1.4

RP4.2	Implement Wi-Fi communication between RPi and PC	Ensure that RPi can be connected to the PC/Laptop over Wi-Fi.	3 days	RP4.1
RP4.3	Bluetooth communication between RPi and tablet device	Ensure that RPi can be wirelessly connected to Nexus 7 tablet device using Bluetooth.	3 days	RP4.1
RP4.4	RPi image recognition	RPi can identify images.	6 days	RP4.1
RP4.5	Implement USB connection between RPi and Arduino board	RPi can communicate and exchange information with an Arduino board using USB connection.	3 days	RP4.1
Android Development (AND5)				
AND5.1	Functional GUI to initiate scanning, selection and connection with Nexus 7 tablet	Application feature that allows for Bluetooth connection between Samsung tablet and the robotic system.	3 days	PM1.4
AND5.2	Bluetooth connection between robot and AA	Android application (AA) can transmit and receive text strings over Bluetooth.	5 days	AND5.1
AND5.3	Implement GUI for the robot system	Ensure robot system status is clearly displayed. Insert input field for waypoint and robot start coordinates.	5 days	RSA2
AND5.4	Implement GUI for the maze environment	Develop GUI for 2D display of maze and robot location, as well as GUI to switch between manual and auto-updating of maze. Display Block number ID in the grid map.	5 days	RSA2
AND5.5	Support user configurable string commands to the robot system	Create 2 buttons to allow custom string commands to be sent to the robot.	2 days	RSA2
Algorithm Development (ALD6)				
ALD6.1	Research on algorithms to implement	Research several possible algorithm options like left-wall hugging etc.	2 days	PM1.4

ALD6.2	Arena exploration simulator	Build a simulator for the arena for MDP to test various aspects.	5 days	ALD6.1
ALD6.3	Map descriptor generation	Implement a file format based on Map Descriptor Format specification to capture arena.	4 days	ALD6.1
ALD6.4	Exploration algorithm development	Develop maze exploration algorithm to explore maze in the shortest time.	7 days	ALD6.1
ALD6.5	Develop shortest path algorithm and simulator	Develop an algorithm for the robot to find the shortest path, and a simulator to test it out.	7 days	ALD6.1
ALD6.6	Time and coverage-limited exploration simulation	Develop a simulator to simulate the algorithm within time and coverage constraints.	5 days	ALD6.1 – ALD6.5
Systems Integration Testing (SIT7)				
SIT7.1	Sub system integration & testing	Ensure that the robot works by integrating all sub systems	5 days	PM1-ALD6
Task Assessment Phase (TA8)				
TA8.1	Week 8 Fastest Path Task Assessment		1 days	SIT7
TA8.2	Week 9 Image recognition Task Assessment		1 days	SIT7
TA8.3	Week 10 Exploration Task Assessment		1 days	SIT7
Documentation Finalization (DF9)				
DF9.1	Project Wiki & Documentation	Document process and technical aspects of the project	40 days	PM1.4
DF9.2	Video Report	Present learning outcomes and achievements of the project through a video report.	5 days	PM1.4

PROJECT ORGANISATION



Position	Name	Tasks
Project Manager	Datta Anusha	• Ensure the team stays on schedule through regular follow-ups and deliverable submission • Boost team morale and help resolve conflicts and when the need arises • Promote healthy discussion, evaluate alternatives and make informed decisions • Ensure team Wiki is updated at all times • Take note of team member contribution and warn slacking members
Algorithm Developers	Bhatia Ritik Datta Anusha	• Implement algorithms that will enable the robot to explore and traverse maze in the fastest time possible, with max time limit being 6 minutes. • The algorithm should have image recognition abilities. It should identify the images it encounters, from a set of 15 images, along with their coordinates. • The algorithms should be able to go through a point with unknown coordinates (waypoint) compulsorily while traversing the fastest path. • The algorithm should allow the robot to return to the initial point once maze exploration is complete. • The algorithm should ensure that the robot avoids obstacles along the way while traversing the maze.
Android Developers	Jeremy Lee Kian Kiat	• Develop and Android application that works with Nexus 7 tablet provided • The application should have the ability to initiate and persist Bluetooth communication with Raspberry Pi

	Kang Wei Bin	● The AA must display grid map, obstacles, explored/unexplored paths and robot's status in real time. ● Implement a functional GUI for the interactive control of the robot movement. ● Develop a GUI that enables input waypoint and start coordinates of the robot.
Raspberry Pi Developers	Quah Dian Wei Irvin Ting Sie Ze	● Develop main communication interface connecting all other aspects and components of the robot system ● Ensure that the camera and the RPi together are able to recognize the images / signs placed on the top of the various obstacles scattered in the map ● Ensure all communication channels setup are secure, uninterrupted & constant, to allow smooth integration. ● Implement RPi access using a PC/notebook over Wifi, can coordinate with Algorithms team for the same
Arduino Developers	Hoy Di Sheng Max Nur Insyirah Lukeman	● Using the hardware provided in the first lab, complete the hardware setup of the robot by week 2 ● Complete hardware testing of the robot by mid week 4 and replace any faulty components hence found ● Perform repeated motor & sensor calibration to ensure reliable and accurate results of distance recorded ● Coordinate and communicate with the RPi t complete communication setup between RPi and Arduino

APPROACH

Strategy

Our group will adopt an Agile Methodology strategy which requires an iterative approach that ensures consistent and frequent deliverables and tasks. With this strategy, different sub-teams are able to perform its different components and tasks at the same time. We aim to hold at least a stand-up weekly meeting during MDP lab sessions where each of the sub-teams can update on their progress and set new targets for the following week. During these meetings, we will ensure that the progress of each sub-teams are consistent and the targets are met as determined by the deliverables and then hold one another accountable for the deliverables.

Integration

During integration, ad-hoc meetings between each sub-teams are done to collaborate on integrating components of different subteams. For example, the Android and Rpi sub-teams will hold ad-hoc meetings outside of the weekly meeting to ensure that the Bluetooth connection works between our two components. This means that the whole team is not required to attend, hence an efficient approach as it does not require the full manpower, allowing the other sub-teams to perform their own work.

Milestones

In this project plan, we have set up a timeline with multiple milestones, which includes some high-priority tasks such as robot assembly as well as checkpoints to ensure consistent progress throughout the course of this project. Each milestone will be based on our deliverables met during the weekly meetings. However, in any case if one sub-team fails to meet its deliverables, we will allocate manpower resources. This flexible adjustment hence creates an efficient way to tackle challenges.

RISK MANAGEMENT

Risk Description	Mitigation/Contingency Plan	Criticality
Hardware components are not functioning properly	• The team is split into sub-teams. Each sub-team is to ensure that their components are functioning by Week 4 so that the team can replace them with the lab supervisor if they are malfunctioning. • If a component is found to be faulty, the sub-team is to immediately inform the Project Manager to ensure that the component gets replaced as soon as possible.	High
Lack of Familiarity with New Packages	• The lab supervisors have conducted a briefing of the project and the 4 components involved. • If a team member is still unsure about the certain aspects of the project, he/she should approach the lab supervisor to seek assistance. • If the team member is still unsure after seeking assistance from the lab supervisor, he/she is to immediately voice out to the Project Manager so that the other members can step in to assist.	High
Conflict of Issues & Miscommunication	• Conduct scrum meetings regularly to check progress, clarify doubts and raise concerns. • Every member is encouraged to immediately voice out concerns, questions, doubts and issues they have at any one point during the project. • Every member is also encouraged to not dismiss issues from other members and to find ways to address/mitigate the issues that were brought up.	High
Timetable Clashes	• The team members will utilize both full sprints and sub sprints to maximize the time spent together as a full team during MDP and time spent between each subteams outside of the allocated time for MDP.	Medium
Lost Components	• The sub-team that requires the components will be in charge of taking care of it. For example, the Raspberry Pi team will be taking care of the Raspberry Pi components and so on. • All sub-teams are to immediately report any missing components or parts immediately to the rest of the subteam especially after Week 4, so that	Medium

	the team can fork out money to replace missing components.	
Freeloaders	• This project allows us to evaluate other team members based on their commitment to the project. The evaluation will be done twice during the project, where the first evaluation will be at Week 6 and the second evaluation will be at Week 12. • The evaluation at Week 6 aims to flesh out any potential issues with team members early on during the project so that the lab supervisors can step in to address the issue. • If the issue still persists even after the lab supervisors have stepped in, the rest of the team will bring it up to the course coordinator and lab supervisors to ensure a fair grade is given to all.	Medium
Looming Deadlines or Inability to Complete Milestones in Time	• Each sub-team consists of two team members. • If a sub-team requires more resources, they will voice out during our full sprints and the team will discuss with the Project Manager.	Medium
Algorithms		
Changing of distance detected by the sensor due to the instability of the sensor. This could lead to the incorrect detection of obstacles and path ahead	• Continual testing and updating required algorithm to alleviate the instability of the sensor	High
Android		
Methods created in Android Studio did not work as intended	• Finish all the required functionalities. • Perform testing on functionalities to eliminate occurrences of bugs	High
Android tablet have low power during deployment	• Fully charge Android battery the day before	Low
Arduino		

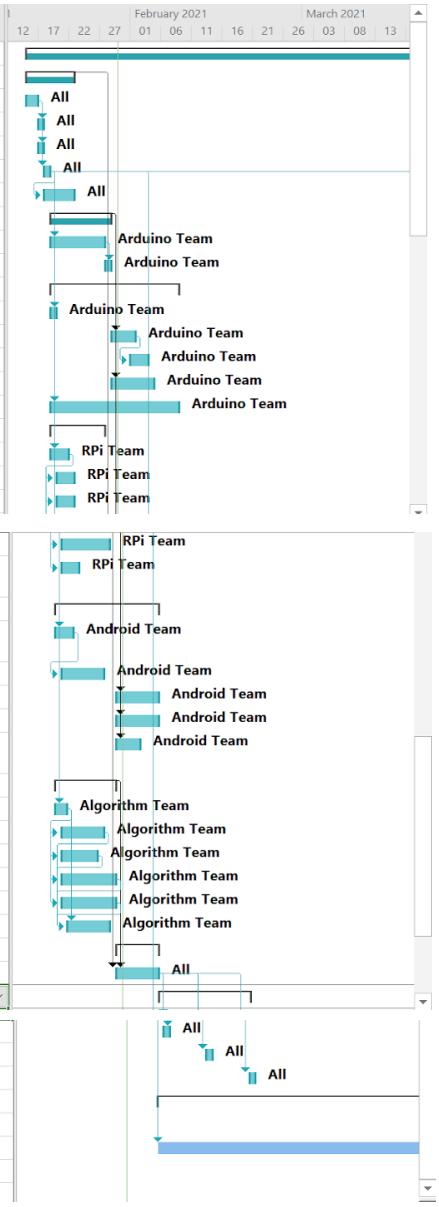
Motor and sensor is defective during the day of test/deployment	• Test and calibrate the sensor and motor continuously. • Report to lab supervisors if the components are defective as soon as possible.	High
Arduino not turning on / not	• Check if the battery is low • If there is power, report it to lab supervisors. • Ensure that the Arduino is fully charged the day before deployment.	High
Raspberry Pi		
The model created with OpenCV uses high computing power, which may slow other processes in RPi	• Try other machine learning libraries in TensorFlow or other deep learning models which are more efficient.	Medium
Lack of coordination between RPi Arduino	• Check if RPi is properly connected to the Arduino. • Check the scripts to ensure that the program is functioning as required.	High

SCHEDULE/TIME MANAGEMENT

Milestones	Estimated Completion
Allocate roles and responsibilities to team members	End of Week 2 (22/01/2021)
Complete Project Plan Documentation and a detailed Work Breakdown Structure for everyone	Mid of Week 4 (03/03/2021)
RPI Operating System (Raspbian) reformatting and configuration along with Wi-Fi/Bluetooth Integration	End of Week 4 (05/02/2021)
Finish configuration of RPI camera and image recognition algorithm	End of Week 5 (21/02/2021)
Complete Android Prototype, remote control and GUI of the grid map	End of Week 7 (26/02/2021)
Finalize simulator of algorithm and navigation algorithm	End of Week 7 (26/02/2021)
Complete implementation and integration of all code on hardware (the robot)	End of Week 7 (26/02/2021)
Finalize all items on Systems Checklist	End of Recess (05/03/2021)
Complete integration of Arduino and hardware with algorithms with fully functioning setup	End of Recess (05/03/2021)
Optimise performance of RPi, Image Recognition, Algorithm and robot	Mid of Week 10 (24/03/2021)
Finalize of Project Wiki and Documentation	Mid of Week 11 (31/03/2021)
Finalization of video report	Mid of Week 11 (31/03/2021)

PROJECT SCHEDULE - GANTT CHART

	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1	**MDP**	**54 days**	**Fri 15-01-21**	**Wed 31-03-21**		
2	**Project Planning/Management**	**6 days**	**Fri 15-01-21**	**Fri 22-01-21**		
3	Project Plan Discussion	2 days	Fri 15-01-21	Sat 16-01-21		All
4	Delegation of Roles & Responsibilities	1 day	Sun 17-01-21	Sun 17-01-21	3	All
5	Environment Setup	1 day	Sun 17-01-21	Sun 17-01-21	3	All
6	Analyse Project Deliverables	1 day	Mon 18-01-21	Mon 18-01-21	3	All
7	Learning/ Knowledge Acquisition	5 days	Mon 18-01-21	Fri 22-01-21	6	All
8	**Robotic System Assembly**	**8 days**	**Tue 19-01-21**	**Thu 28-01-21**		
9	Hardware Components Collection and Testing	7 days	Tue 19-01-21	Wed 27-01-21	6	Arduino Team
10	Assemble Robot	1 day	Thu 28-01-21	Thu 28-01-21	9	Arduino Team
11	**Arduino Development**	**15 days**	**Tue 19-01-21**	**Mon 08-02-21**		
12	Set up Arduino Environment	1 day	Tue 19-01-21	Tue 19-01-21	6	Arduino Team
13	Motor Calibration	2 days	Fri 29-01-21	Mon 01-02-21	8	Arduino Team
14	a	3 days	Mon 01-02-21	Wed 03-02-21	13	Arduino Team
15	IR Sensor Calibration	5 days	Fri 29-01-21	Thu 04-02-21	8	Arduino Team
16	Main code for Arduino	15 days	Tue 19-01-21	Mon 08-02-21	6	Arduino Team
17	**Raspberry Pi Development**	**7 days**	**Tue 19-01-21**	**Wed 27-01-21**		
18	Raspberry Pi Setup	3 days	Tue 19-01-21	Thu 21-01-21	6	RPi Team
19	Implement Wi-Fi communication between RPi and	3 days	Wed 20-01-21	Fri 22-01-21	18	RPi Team
20	Bluetooth communication between RPi and tablet	3 days	Wed 20-01-21	Fri 22-01-21	18	RPi Team
21	RPi image recognition	6 days	Wed 20-01-21	Wed 27-01-21	18	RPi Team
22	Implement USB connection between RPi and Arduino board	3 days	Wed 20-01-21	Fri 22-01-21	18	RPi Team
23	**Android Development**	**13 days**	**Tue 19-01-21**	**Thu 04-02-21**		
24	Functional GUI to initiate scanning, selection and connection with Nexus 7 tablet	3 days	Tue 19-01-21	Thu 21-01-21	6	Android Team
25	Bluetooth connection between robot and AA	5 days	Wed 20-01-21	Tue 26-01-21	24	Android Team
26	Implement GUI for the robot system	5 days	Fri 29-01-21	Thu 04-02-21	8	Android Team
27	Implement GUI for the maze environment	5 days	Fri 29-01-21	Thu 04-02-21	8	Android Team
28	Support user configurable string commands to the robot system	2 days	Fri 29-01-21	Mon 01-02-21	8	Android Team
29	**Algorithm Development**	**8 days**	**Tue 19-01-21**	**Thu 28-01-21**		
30	Research on algorithms to implement	2 days	Tue 19-01-21	Wed 20-01-21	6	Algorithm Team
31	Arena exploration simulator	5 days	Wed 20-01-21	Tue 26-01-21	30	Algorithm Team
32	Map descriptor generation	4 days	Wed 20-01-21	Mon 25-01-21	30	Algorithm Team
33	Exploration algorithm development	7 days	Wed 20-01-21	Thu 28-01-21	30	Algorithm Team
34	Develop shortest path algorithm and simulator	7 days	Wed 20-01-21	Thu 28-01-21	30	Algorithm Team
35	Time and coverage-limited exploration simulation	5 days	Thu 21-01-21	Wed 27-01-21	30,31,32,33,34	Algorithm Team
36	**Systems Integration Testing**	**5 days**	**Fri 29-01-21**	**Thu 04-02-21**		
37	Sub system integration & testing	5 days	Fri 29-01-21	Thu 04-02-21	2,29	All
38	**Task Assessment Phase**	**11 days**	**Fri 05-02-21**	**Fri 19-02-21**		
39	Week 8 Fastest Path Task Assessment	1 day	Fri 05-02-21	Fri 05-02-21	37	All
40	Week 9 Image recognition Task Assessment	1 day	Fri 12-02-21	Fri 12-02-21	37	All
41	Week 10 Exploration Task Assessment	1 day	Fri 19-02-21	Fri 19-02-21	37	All
42	**Documentation Finalization**	**40 days**	**Thu 04-02-21**	**Wed 31-03-21**		
43	Video Report	5 days	Thu 25-03-21	Wed 31-03-21	6	All
44	Project Wiki & Documentation	40 days	Thu 04-02-21	Wed 31-03-21	6	All



Appendix: Key Terms

In the following table, we have provided definitions for terms that are relevant to the document and is intended to provide a clearer understanding of the context in which they are used:

Term	Definition
RPi	Stands for Raspberry Pi, which is a board that acts as a CPU in itself
GUI	Stands for Graphical User Interface and it means the interface that a user interacts with. It is the Point of Contact of user with the application, to use the related functionality of the app
RPM	Stands for Revolutions Per Minute. It means the total complete revolutions made by the wheel in a time period of 1 minute
PID	Stands for Proportional, Integral, Derivative. It is a closed loop system that keeps track of the error accumulated by the motor and makes corrections accordingly so that a desired RPM can be maintained. This is to ensure proper traversal by the robot.
AA	Stands for Android Application. It is the application that communicates with the robot.
IR	Stands for Infrared and is a type of sensor that is used with the Arduino component of the robot